

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student:

Tomáš Blažek

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Absolvování individuální odborné praxe
Individual Professional Practice in the Company

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: AutoCont CZ a.s.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.


Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **doc. Mgr. Jiří Dvorský, Ph.D.**


Konzultant bakalářské práce: Kamila Rašková

Datum zadání: 01.09.2015

Datum odevzdání: 29.04.2016


doc. Dr. Ing. Eduard Sojka
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.


V Ostravě 29. dubna 2016

.....


*Souhlasím se zveřejněním této bakalářské/diplomové práce dle požadavků čl. 26, odst. 9
Studijního a zkušebního řádu pro studium v bakalářských/magisterských programech
VŠB-TU Ostrava*

AutoCont
AutoCont CZ a.s.
Hornopolská 34, 702 00 Ostrava
Tel.: 910 971 111, fax: 910 970 100
DIČ: CZ47676795 ²⁰

V Ostravě dne 22. 4. 2016


.....

Rád bych poděkoval vedení firmy AutoCont za to, že mi tuto praxi umožnilo. Jmenovitě pak panu Miloslavovi Záleskému, ale také celému kolektivu spolupracovníků. Dále bych chtěl poděkovat panu doc. Mgr. Jiřímu Dvorskému, Ph.D za příjemný přístup při odborném vedení a rady, které jsem od něj získal.

Abstrakt

Tato bakalářská práce pojednává o průběhu mé odborné praxe ve firmě AutoCont CZ a.s. v průběhu posledního ročníku mého bakalářského studia. Firma AutoCont se zabývá dodávkou zboží, řešení a služeb na poli IT. Cílem mé práce je popis zadaných projektů, použitých technologií a postupů.

Klíčová slova: JavaScript, jQuery, praxe, Bootstrap, HTML, CSS

Abstract

This bachelor thesis describes the process of my practice in AutoCont CZ a.s. company during my last year of bachelor studies. Autocont is engaged in supplying goods, solutions and services on the IT field. The goal of my thesis is description of the assigned projects, used technologies and procedures.

Key Words: JavaScript, jQuery, practice, Bootstrap, HTML, CSS

Obsah

Seznam použitých zkratk a symbolů	9
Seznam obrázků	10
Seznam tabulek	11
1 Úvod	13
2 Představení společnosti	14
2.1 AutoCont	14
2.2 Popis pracovního zařazení	14
2.3 Firemní standardy	14
3 Technologie	15
3.1 ASP.NET MVC 4 s Razor engine	15
3.2 JavaScript	15
3.3 jQuery	15
3.4 HTML5	16
3.5 CSS3	16
3.6 Bootstrap	16
4 Projekty	18
4.1 Web AutoCont CZ/SK/EN	18
4.2 eKatalog	18
4.3 Charitativní projekt AC25	19
4.4 SharePoint	20
4.5 Registrace na akce firmy AutoCont CZ i SK	21
4.6 AC Cloud	21
4.7 eKatalog Management Studio	22
4.8 Image Editor	25
4.9 IT pro Byznys	28
5 Časová náročnost projektů	30
6 Znalosti	31
6.1 Znalosti načerpané při studiu na VŠB-TUO uplatněné v praxi	31
6.2 Chybějící znalosti při nástupu do firmy	31
7 Závěr	32

Seznam použitých zkratek a symbolů

Ajax	– Asynchronous JavaScript and XML
API	– Application Programming Interface
ASP	– Active Server Pages
CDN	– Content Delivery Network
CSS3	– Cascading Style Sheets verze 3
HTML5	– HyperText Markup Language verze 5
JS	– JavaScript
REST	– Representational State Transfer
SHA1	– Secure Hash Algorithm 1
TFS	– Team Foundation Server
RGBA	– Red, Green, Blue, Alfa
UI	– User Interface
URL	– Uniform Resource Locator
UX	– User Experience
WYSIWYG	– What You See Is What You Get
XML	– Extensible Markup Language

Seznam obrázků

1	Vizualizace rozcestníku	23
2	Přetahování jednotlivých položek pomocí sortable	24

Seznam tabulek

1	Tabulka časové náročnosti jednotlivých projektů	30
---	---	----

Seznam výpisů zdrojového kódu

1	Získávání reference na nově vytvořené objekty	25
2	Práce s RGBA hodnotami v canvas za pomoci JavaScriptu	26

1 Úvod

Pro vypracování bakalářské práce formou praxe jsem se rozhodl, protože jsem chtěl rozšířit své teoretické znalosti o dovednosti z praxe a využít této možnosti k tomu, abych po dokončení studia mohl být více zajímavým pro mé budoucí zaměstnavatele díky zkušenostem a praktickým znalostím nabytým během této stáže.

V této mé práci budu popisovat mou praxi ve firmě AutoCont CZ na pozici Webmaster. Budu zde popisovat projekty, na kterých jsem za svého působení v této firmě pracoval. Dále budu také objasňovat má, nebo týmová, technologická rozhodnutí při přípravě projektů v rámci možností projektového zadání od vedení firmy.

2 Představení společnosti

2.1 AutoCont

AutoCont je česká soukromá společnost, která přes 25 let na českém a slovenském trhu úspěšně zavádí a provozuje užitečné informační technologie. Zaměřuje se na poskytování komplexních IT řešení a služeb pro firemní klientelu a státní správu.

2.2 Popis pracovního zařazení

Oficiální název mé pozice je Webmaster/grafik, ačkoli to ne úplně přesně vystihuje mé zaměření. Mou prací je front-end a částečně i back-end výstup našich webů a microwebů. Jedná se o strukturování, stylování, udržování webů spolu se skriptováním. Grafikou se v mé práci zabývám jen okrajově, spíše je důležité estetické cítění pro strukturování a stylování stránek v duchu firemního designu, který se objevuje napříč weby.

2.3 Firemní standardy

Standardem v našem oddělení je pro back-end práce s ASP.NET MVC s Razor engine. Vizuální stránka je poté tvořena za pomoci HTML5 a CSS3 standardů přičemž pro práci s CSS využíváme CSS procesoru LESS nastaveného pro kompilování v nástroji Microsoft Visual Studio 2012, ve kterém se odehrává veškerá práce. Pro interaktivitu webového prostředí se využívá skriptovacího prototypovacího jazyku JavaScript spolu s frameworky jQuery a jQuery UI. JQuery využíváme i pro asynchronní volání obsahu pomocí AJAX requestů. Dále využíváme HTML5, CSS3 a Javascript frameworku bootstrap pro responzivitu webu. Všechny tyto technologie jsou v dnešní době tzv. industry standards, neboli standardy v našem odvětví. Ke sdílení kódu a jeho archivaci používáme funkci zabudovanou ve Visual Studiu s názvem TFS, neboli Team Foundation Server.

3 Technologie

Zde bych se rád věnoval, při mé práci, nejčastěji používaným technologiím.

3.1 ASP.NET MVC 4 s Razor engine

Jedná se o webový aplikační framework od společnosti Microsoft, který implementuje architekturu MVC, což je zkratka pro tři hlavní komponenty webu Model-View-Controller. Model má reprezentovat data a business logiku, View převádí data modelu do uživatelsky přívětivého rozhraní a Controller se stará o aplikační logiku. Razor View engine umožňuje uživateli pracovat v HTML s jazyky C# nebo Visual Basic a generovat HTML validní kód. Pro uživatele je snadno rozpoznatelný díky absenci přípony v URL (v projektu má příponu .cshtml) adrese na rozdíl od Web Forms, které se vyznačují příponou (.aspx). Jeho syntaxe vždy začíná znakem zavináč.

3.2 JavaScript

Jedná se o prototypový interpretovaný skriptovací jazyk běžící na klientské straně, avšak existují i server-side JavaScript aplikace. Implementují ho všechny nejznámější webové prohlížeče světa. Jeho syntaxe je velmi podobná jazyku Java jak část názvu tohoto jazyku napovídá.

Pomocí tohoto jazyka můžeme tvořit interaktivní webové stránky, aplikace či dokonce hry. V poslední době je velká snaha interaktivitu webu převést spíše na HTML a CSS, avšak mnoho starších prohlížečů nemá implementované nejnovější standardy těchto jazyků a tak je prozatím JavaScript stále důležitou součástí uživatelského zážitku.

Budoucnost tohoto jazyku se v posledních letech ukázala ve spolupráci s HTML elementy canvas a svg. Umožňuje díky nim vytvářet hry nezávislé na platformě Adobe Flash, která ještě donedávna dominovala webovým aplikacím a hrám.

JavaScript se v poslední době nejčastěji vyskytuje spolu s frameworky na něm postavenými jako je AngularJS, jQuery, Node.js a jinými.

3.3 jQuery

Je nejrozšířenějším open-source frameworkem postaveném na JavaScriptu. Jedná se o velmi malý soubor, což je jedna z velkých výhod jQuery. Implementuje Sizzle selektovací engine, který naprosto změnil styl programování v JavaScriptu. jQuery je hojně využíváno také pro práci s Ajax, kterou má velmi dobře implementovanou. Syntaxe jQuery je složena ze selektoru, který je složen ze znaku dolaru následujícím id, názvem třídy či jménem HTML tagu obaleném v uvozovkách a závorkách. Následně je vložena tečka pokračující názvem metody, kterou chceme volat na vybraný element. V závorkách jsou pak vloženy parametry metody. Metody se dají řetězit tečkami za sebou. Framework umožňuje zapojit i event listenery v takovém případě se do parametrů zadává buďto jméno funkce, která se má při spuštění eventu zapnout, nebo se dá tato funkce přímo vepsat mezi závorky.

3.4 HTML5

Je nejnovějším standardem značkovacího jazyku HTML. HTML má za úkol vytvořit strukturu webu, která je následně doplněná stylováním CSS. S touto verzí přišly nové elementy jako například canvas pro práci s rastrovou grafikou, multimediální elementy typu video a audio či elementy pro strukturování header, footer atp. Poslední zmíněné jsou velmi důležité pro SEO optimalizaci jelikož algoritmy vyhledávačů velmi přihlíží k rozdělení obsahu pomocí těchto elementů. Syntaxe HTML5 se vyznačuje špičatými závorkami uvnitř kterých je napsáno jméno tagu a jeho atributy jako je například id, class atp. Párové tagy se ukončují ještě jedněmi špičatými závorkami s jménem tagu uvnitř, kterému předchází lomítko. Mezi těmito tagy se nachází obsah, který by měl tento element v sobě obsahovat. Existují i nepárové tagy, které se oproti (X)HTML neukončují v prvním tagu lomítkem před druhou špičatou závorkou.

3.5 CSS3

Kaskádové styly jsou nedílnou součástí webu. Slouží pro stylizování struktury vytvořené v HTML pomocí selektorů jako je id, jméno třídy nebo jméno prvku. Určuje barvy prvků, styly písma, ohrazení, zalamování prvků a mnoho dalšího. Nová verze s názvem CSS3 sebou přinesla mnoho užitečných dlouho očekávaných atributů jako je zaoblení rohů, stíny textů apod.

Syntaxe je jednoduchá. Vždy se začíná selektorem, který je stejný jako v jQuery tzn. jméno třídy následuje tečku, id následuje mřížku a jména tagů jsou bez znaku. Tyto selektory se za sebou řadí podle vnoření v HTML. Za selektorem následují složené závorky, ve kterých definujeme hodnoty atributu pomocí názvu atributu následovaném dvojtečkou a samotnou hodnotou ukončenou středníkem.

Velmi důležitou součástí CSS jsou takzvané media queries, která se využívají pro responzivitu designu. V bloku media queries se píše kód stejně jako vně, ale nemusí se vypisovat celý znovu, ale pouze ty atributy, které chceme pozměnit pro určitý selektor. Tento blok je definován nejčastěji maximální, či minimální šířkou nebo výškou viewportu neboli průhledu v pixelech, dpi. Možností je však více např. nastavit jiný design pro tisk atp.

3.6 Bootstrap

Jedná se o open-source front-end framework který obsahuje HTML a CSS designy spolu s resetovacími kódy, ale také JavaScript rozšíření. Jeho dělení stránky do sloupců, kterých je v základu dvanáct, ale dá se nastavit i jiný počet, umožňuje vývojáři pohodlně tvořit responsivní mobile-first web.

Sloupce layoutu se nastavují pomocí tříd se jménem začínajícím zkratkou velikosti v angličtině („xs“ pro extra small, „sm“ pro small, „lg“ pro large atp.) následující pomlčkou a řadou znaků „col“, z anglického column, které je dále následováno číslem určujícím jak velkou část z např. 12 sloupcového layoutu má tento prvek zabrat.

Já osobně, ačkoli jsem byl nucen s Bootstrapem pracovat, nejsem jeho velkým zastáncem. Dle mého názoru jde o příliš velkou knihovnu, která se dá jednoduše nahradit vlastním CSS. Na druhou stranu chápu její oblíbenost mezi vývojáři, protože dovoluje začínajícím kódérům urychlit vývoj webu, bez nutnosti rozvrhnout si dopředu jeho strukturu, nebo hlubší znalosti HTML a CSS.

4 Projekty

4.1 Web AutoCont CZ/SK/EN

4.1.1 Zadání

Potřeby marketingového oddělení byly jasné a stručně popsány na poradě při mém vstupu do firmy. Bylo zapotřebí renovovat oficiální web firmy AutoCont, který ani zdaleka neodpovídal dnešním standardům na responzivní strukturu, pro všechny jazykové verze. Důraz byl kladen na úvodní stránku, kde se nachází dlaždice s obrázkovými slidery uspořádané v responzivní formě.

4.1.2 Technologická rozhodnutí

Vzhledem ke firemním standardům jsme využili frameworku Bootstrap pro strukturování responzivního webu, jQuery a jQuery UI pro interaktivní uživatelské rozhraní, CSS procesoru LESS a ASP.NET MVC pro backend samotného webu. Z naší strany byly k zadání vzneseny návrhy na využití SQL databází, ale byly zamítnuty. Proto jsme některé funkcionality starého webu zachovali a pracovali s daty uloženými v XML souborech.

4.1.3 Problémy

Problémy pro naši, jakožto front-end vývojářskou stranu, nastaly s dílčími úpravami ze strany vedení. Tyto úpravy nebyly očekávány při tvorbě návrhu a toto vedlo k nepříjemným řešením v podobě mnohonásobně většího počtu šablon, které byly navíc rozdílné pro každou jazykovou skupinu.

4.1.4 Hodnocení

Web byl ze strany vedení přijat pozitivně a na jeho úpravách se stále pracuje. Za mne mohu říci, že deadline, podle kterého jsme se museli řídit a který nás nutil do nejrůznějších kompromisů je na tomto projektu citelně vidět. Mimo jiné bych nyní celou řadu problémů řešil jinak, díky mému pokroku zejména v oblasti jQuery, potažmo JavaScriptu.

4.2 eKatalog

4.2.1 Zadání

Firma AutoCont dlouhodobě vydávala svůj katalog produktů v tištěné podobě, avšak dostala se k bodu, kdy padlo rozhodnutí o přemístění tohoto katalogu na web. Mělo se jednat o web, který bude responzivní, bude si držet formu a bude lehce obsluhovatelný ze strany uživatele. Projekt měl být součástí webu AutoCont a to jak v českém jazyce, tak i slovensky. Úkol byl delegován mi, jakožto můj první osobní projekt.

4.2.2 Technologická rozhodnutí

AutoCont zadal vypracování vizuálu grafickému studiu Aetna, která posléze dodala svůj minimalistický návrh. Grafický návrh bylo potřeba překlomit do responzivní formy HTML a CSS a navíc domyslet části, které v grafice nebyly vizualizovány (mezikroky v responsivitě, dodání ikon... atp.).

4.2.3 Problémy

Jakožto nezkušený v oblasti responsivity, jsem na problémy narazil záhy a to v podobě grafického návrhu, který s ní nijak zásadně nepočítal. Jediným dalším problémem, který se objevil až po čase, byla složitost HTML struktury na osobu, která se měla starat o obsah. Toto mělo za následek křížení tagů, nebo záměna jmen CSS tříd.

4.2.4 Hodnocení

Spokojenost s webem byla v době jeho dokončení, jak z mé, tak ze strany vedení velmi vysoká. Nyní však nastala, díky problémům s udržováním obsahu, potřeba změny. Proto jsem se rozhodl pro vytvoření velice přívětivého a intuitivního grafického rozhraní pro správu obsahu eKatalogu.

4.3 Charitativní projekt AC25

4.3.1 Zadání

AutoCont má na Vánoce ve zvyku dávat svým partnerům nějaké malé dárky. Tento rok (2015) se ale rozhodl díky svým 25. narozeninám darovat něco, co pomůže a to formou charitativního voucheru. Byl jsem osloven, zda bych se tohoto projektu nechtěl ujmout, což jsem s radostí udělal. Toto rozhraní mělo běžet na stránkách www.autocont.cz/25.

Mým úkolem bylo vytvořit "účty" uživatelů a webové rozhraní, kterými budou přispívat na dobročinné účely, dle jejich uvážení. Na výběr byly čtyři možnosti výběru. Každý voucher měl hodnotu 250,- Kč a bylo jich vyrobeno 4500 tudíž jsem měl na starost projekt o hodnotě 1.125.000,- Kč. Toto byla první fáze projektu.

Druhá fáze byla zaměstnanecká sbírka, kdy si zaměstnanec mohl vybrat sumu, která mu bude strhnuta z platu a dána na jednu z charitativních možností, které byly stejné, jako v první fázi určené pro AutoCont partnery. Tato fáze neměla probíhat paralelně, ale až okolo nového roku.

4.3.2 Technologická rozhodnutí

V první fázi bylo důležité rozhodnout se, jak moc zabezpečené tyto účty mají být. Bylo potřeba vygenerovat číslo voucheru a bezpečně jej uložit šifrováním do xml souboru. Navrhoval jsem aby číslo voucheru obsahovalo malá i velká písmena, čísla a znaky a to v délce nejméně deseti symbolů. Tento návrh byl zamítnut s tím, že našim partnerům se nebude chtít zapisovat sáhodlouhé

bloky znaků a byl jsem požádán, ať se jedná o co nejmenší kód, začínající na "AC25" obsahující jen velká písmena a čísla. Mé námitky na počet kombinací a nebezpečnost byly zamítnuty. Proto jsem vytvořil hesla o délce 8mi znaků z čehož 4 byly výše zmíněné "AC25". To činí 1.679.616 možností hesel. Pro zakódování hesla jsem použil SHA1 a Salt.

Druhou fází jsem chtěl provázat s firemní databází zaměstnanců v Microsoft AX, avšak vzhledem ke složitosti a deadline projektu se řešil tento proces pomocí vkládání zaměstnaneckých kódu a následným odesláním přes back-end webu na emailové adresy účtárny. Jednalo se tak vzhledem k obavám zaměstnavatele o údaje. K těmto údajům by mělo přistup více lidí, pakliže by byly na firemním serveru a tímto by mohlo dojít k únikům dat.

4.3.3 Problémy

Chyba v malém počtu znaků hesla, absence case sensitivity a znaků měla za následek několik nespokojených lidí, kteří nemohli přispět. Tento problém nastal z očekávaného malého počtu možností hesel. Proto bylo pro některé vtipálky jednoduché tipnout si heslo a hlasovat za někoho jiného. Avšak tento problém nebyl velkých rozměrů a z 4.500 lidí mělo tento problém maximálně 5 z nich.

4.3.4 Hodnocení

I přes menší nepříjemnosti v první fázi projektu byl zadavatel spokojen a vše proběhlo v pořádku. Do zaměstnanecké části jsem již po vypracování nezasahoval, ale dle mých informací vše proběhlo v pořádku a vedení bylo velmi spokojeno.

4.4 SharePoint

4.4.1 Zadání

Práce s portálem SharePoint firmy Microsoft, byla jedna z věcí náplně mé práce, kdy jsem se měl naučit práci s administrací tohoto interního portálu. Jednalo se o vytváření sekcí, menu, podpoložek, uploady souborů, jejich publikování a práce s online zprostředkováním videí z našeho serveru.

4.4.2 Problémy

Vzhledem k tomu, že jsem se v mé práci s portálem SharePoint setkal poprvé, byl jsem velice nemile překvapen. Ačkoli se jedná o nástroj firmy Microsoft, byl velice nestabilní a nekonzistentní na všech prohlížečích. Při práci s videi se často zasekával či úplně shodil prohlížeč.

4.4.3 Hodnocení

Po projednání mých námitek s vedením, mi (a mému mentorovi) bylo dáno částečně za pravdu a jsou plánovány nové alternativy do budoucna.

4.5 Registrace na akce firmy AutoCont CZ i SK

4.5.1 Zadání

Vyvstala potřeba pro online registrace na akce firmy AutoCont CZ a AutoCont SK, jako je každoroční akce Techforum, které se účastní jedny z největších firem na poli IT. Za úkol jsme měli vymyslet logické řazení paralelních rozvrhů s důrazem na délku obsahu stránky a znovupoužití.

4.5.2 Technologická rozhodnutí

Vytvořily se metody na back-endu naší serverové aplikace pro kontrolu vyplněnosti, správnosti registračních údajů, odeslání mailů a udržování dat. Řešení onoho důrazu na krátkost obsahu, jsme se rozhodli řešit pomocí JavaScript, přesněji jQuery UI za pomoci rolovacích zobrazení podrobností.

4.5.3 Hodnocení

Jak česká tak slovenská strana byla velmi spokojena, následně jsme se dozvěděli, že díky vytíženosti sítě v místě konání akce, kde se lidé připojovali svými smartphony a prohlíželi si rozvrh, AutoCont získal zakázku na síťovou podporu daného hotelu.

4.6 AC Cloud

4.6.1 Zadání

Jedná se o jeden z větších webu firmy AutoCont, kde se nabízí cloudové služby. Byly nám dodány grafické podklady grafického studia Aetna, podle nichž jsme měli vytvořit tento web. Tak jako u předchozích projektů byla požadována responsivita, dobré UX a využití JavaScriptu. Back-end webu měl obsluhovat objednávací, později informativní, formulář a umožnit práci s XML soubory.

4.6.2 Technologická rozhodnutí

Při tvorbě tohoto webu bylo zapotřebí nejen CSS3 funkcí, ale také ve velké míře JavaScript s frameworky jQuery a jQuery UI. Samozřejmě jako u všech webů, o kterých v této práci referuji i ASP.NET MVC s Razor. V tomto projektu bylo i využito Ajax volání pro spojení s Administrativním registrem ekonomických subjektů.

4.6.3 Problémy

Ačkoli tento web, pokud se na něj podíváme na papíře, nevypadal nikterak složitě, opak byl pravdou. Bylo zde hodně problémů s kompatibilitou na všech nejpoužívanějších prohlížečích a to jak v CSS, tak v JS. Dále zde bylo několik web funkcí, které se nakonec do finální verze

nedostaly, jako neredundantní výpis formulářů, místo kterého mělo být Ajax volání vnitřního jednotného html souboru. Ačkoli tato funkce byla plně funkční, nakonec nebyla zapotřebí.

4.6.4 Hodnocení

Na tomto webu jsme strávili asi nejvíce času ze všech webů, na kterých jsme pracovali týmově. Při řešení JavaScript problémů jsem se zde naučil nejvíce ze všech týmových projektů a řekl bych, že to pro mne byl zlomový okamžik, kdy jsem se plnohodnotným článkem našeho týmu. Zjistil jsem, že JavaScriptu, nebo spíše jQuery, začínám opravdu rozumět a jsem schopný řešit problémy, které mi dříve trvaly dny, během několika minut. Ačkoli jsem při začátcích ještě neřešil přehlednost kódu, nebo jeho znovupoužití, ke konci projektu jsem již pochopil výhodnost těchto postupů. Řekl bych, že tento projekt mě zformoval a ukázal, na co se chci do budoucna specializovat.

4.7 eKatalog Management Studio

4.7.1 Zadání

Zadáním bylo vytvořit UI administraci pro spolupracovníky v marketingovém týmu, které by umožňovalo realtime upravovat stránku v duchu WYSIWYG. Dále bylo zapotřebí oddělit vedoucí a administrátory od ostatních zaměstnanců. Toto bylo z důvodu umožnění, pouze určitým lidem, uploadu upraveného HTML souboru přímo na ostrý web. Dále byl požadavek na historii úprav sekcí a práci na posledním upraveném HTML souboru. Tento požadavek byl vznesen z důvodu častých opakování některých produktů v dalších číslech eKatalogu. Toto byly základní požadavky pro testovací provoz. Dále by byl přidán upload obrázků a plánuje se eKatalogu přidat funkcionality eshopu.

4.7.2 Technologická rozhodnutí

Rozhodnul jsem se tuto administraci zakomponovat do našeho stávajícího webu, abych se vyhnul možným problémům s přenášením dat pomocí AJAX z webu na web, tudíž bylo použito ASP.NET MVC s Razor.

Autentizaci řeším na Login stránce, kde klasicky zasílám uživatelské jméno a heslo do backend, kde jej šifruji za pomoci SHA1 a Salt a kontroluji jeho shodu s výskytem v XML souboru. Všechny ostatní stránky, které jsou mezi View této administrace, jsou při neautentizovaném přístupu přesměrovány na stránku Login pomocí routingu ve webconfigu.

Po přihlášení jsem se rozhodl vytvořit rozcestník, který obsahuje jména kategorií eKatalogu jejich historii přístupu a také kontrolu uzamčení sekce jiným uživatelem. K tomuto uzamčení dochází ihned při kliknutí uživatele na tlačítko "edit" dané sekce. Díky tomuto je zajištěno, že si uživatelé nebudou nechtěně přepisovat své úpravy. Tlačítko bude znepřístupněno, pakliže je tuto sekci již někdo edituje. Při rozkliknutí historie sekce se stránka posune k výpisu z historie, který

je volán za pomoci AJAX requestu. K AJAX requestu jsem se přiklonil, jelikož není důležité tyto data mít v prohlížeči, časem by mohly být i zátěží na rychlost načtení stránky, pakliže na ně uživatel neklikne. Volá se pouze historie jedné sekce, kterou má uživatel zrovna otevřenou.

The screenshot displays the 'eKatalog management studio' interface. On the left is a dark red sidebar with the logo and a 'log out' button. The main area features a table with categories and their history. Below this is a table titled 'rozcestníku' (breadcrumbs) showing recent navigation paths.

From	To	Name	Type
7. 4. 2016 15:00	7. 4. 2016 15:35	Kamila Rašková	Published
5. 4. 2016 12:00	6. 4. 2016 16:00	Tomis Miloš	Edited
4. 3. 2016 10:17	4. 3. 2016 11:35	Tomáš Blažek	Edited

Obrázek 1: Vizualizace rozcestníku

Při otevření sekce jsem využil načítání dat z XML souboru do HTML stránky. Spustí se Javascript, který je dělen do main.js, edit.js, drag.js, pop-up.js a template-buttons.js. Jak názvy napovídají, tak main slouží jako inicializace a základní funkcionalita, nezávislá na editaci. Edit soubor obsahuje funkcionalitu pro realtime editaci textů, výpočet DPH či potvrzování úprav. Soubor drag slouží pro zajištění funkcionality přesouvání položek mezi sloupci. JavaScript soubor pop-up zajišťuje jak zobrazení pop up okna, tak i jeho vnitřní funkcionalitu. Uvnitř pop-up okna můžeme najít možnosti nastavení odkazů (zda-li se jedná o email, odkaz na vnitřní web či vnější), ale také úpravu formátu ceny, upload obrázku a také se pracuje na možnosti editace obrázku. Template-buttons má na starost funkce tlačítek v horním menu, jako je přidávání položek,

mazání, kontrola módu (drag nebo edit), ale také dává možnost vybrat barvu a stranu, na které se zobrazí sloupec se zvýrazněnou nabídkou. Potřebu rozlišení mezi obyčejným uživatelem a uživatelem s vyššími právy jsem řešil právě pro toto view, kdy je potřeba zajistit možnost publikace na web pouze pro uživatele s vyššími právy. Toto jsem řešil pomocí cookies a ukládání role uživatele, která je následně kontrolována nežli se vygeneruje kód.



Obrázek 2: Přetahování jednotlivých položek pomocí sortable

4.7.3 Problémy

Jakožto můj zatím největší samostatný projekt jsem zde narazil na mnoho problémů. Ať už s JavaScriptem, či .NET. JavaScript této stránky jsem celý tři krát přepisoval díky novým znalostem, které jsem během psaní prvních dvou verzí nabyl. Velkým problémem bylo nastavit editovatelné texty, aby rozlišily, zda se mají vyměnit za input s atributem type nastavený na text, či na textarea. Se samotným nastavováním funkcí na textarea nastaly další problémy, jelikož textarea není automaticky zvětšovatelná podle velikosti textu, což se muselo řešit event listenerem, který hlídal počty řádků a jejich výšku s výškou samotného tagu textarea, při každé změně. Nakonec jsem nebyl spokojen se složitostí textových úprav, jako je ztučnění písma, či podtrhnutí, pro které by byla nutná znalost HTML, tudíž by se má administrace nestala user friendly. Proto jsem po pečlivém prostudování možností rozhodnul použít tagu div s atributem „contenteditable“. Toto mi dovolilo využít funkce document.execCommand(), díky které jsem mohl jednoduše text formátovat. Dalším problémem bylo získání reference nového elementu vytvořeného po akceptování úprav. Po dlouhých úvahách a studiu na internetu jsem narazil na možnost využití zachycení eventu DOMNodeInserted, díky čemuž jsem mohl pracovat s referencemi na nově vytvořené DOM objekty. Při práci na back-endu aplikace jsem narazil na

nespočet problémů v podobě velmi rozsáhlé struktury stavajícího AutoCont webu. Bylo nutno přeroutovat cesty, nastavit webconfig separátně pro různé oblasti webu a pozměnit nastavení views pro mou část webu.

```
//Pri vložení DOM objektu odpovídajícímu druhému parametru spustit funkci a
    předat event
$(document).on('DOMNodeInserted', '.column-item>p, .column-item>ul', function (
    e) {
    // Uložit do promenne rodicovsky prvek
    var colItem = $(e.target).parent();
    //aktivovat event listener na funkci click
    $(this).click(function () {
        //zavola se funkce clickOnEditable, která objekt znovu meni na
        upravovatelný objekt
        clickOnEditable(e.target, colItem, true);
    });
});
```

Výpis 1: Získávání reference na nově vytvořené objekty

4.7.4 Hodnocení

Zprvu se zdálo, že můj nápad na tuto administraci se setkává s negativním ohlasem, ale po té co jsem vysvětlil výhody a ulehčení práce všem v kolektivu se pro to mnoho lidí nadchlo. Lidé se často ptali, kdy budou moci přispívat do nové administrace a přicházeli s různými nápady, jak tento projekt vylepšit. Jednou z těchto věcí je i image editor, o kterém píšu níže. Já osobně mám z tohoto projektu dobrý pocit, jelikož mne posunul skokem o mnoho dále v mých vědomostech.

4.8 Image Editor

4.8.1 Zadání

S tímto nápadem za mnou přišel kolega, který má na starosti fotky pro eKatalog. Ptal se, zda-li by bylo možné implementovat do eKatalog Management Studia aplikaci pro odstraňování pozadí z fotografií bez potřeby přesného ořezu v grafickém programu.

4.8.2 Technologická rozhodnutí

Vzhledem k tomu, že projekt takového typu byl pro mne novinkou, bylo potřeba prozkoumat možnosti jak takovýto projekt řešit. Důležité bylo se rozhodnout zda se bude jednat o online řešení, nebo desktopovou aplikaci. Zde jsem se rozhodl pro online cestu, jelikož bylo velice důležité, aby tato aplikace byla rychle přístupná přímo při práci na sazbě eKatalogu. Online

aplikace mohla být řešena na front-end, či back-endu. Od zpracování na serveru jsem po uvážení ustoupil, vzhledem k potřebě znát přesné údaje pro ořez bez možnosti živého náhledu, při kterém by se tyto data daly získat.

Proto jsem přistoupil k řešení za využití HTML5 elementu canvas, který umožňuje za pomoci JavaScriptu živě pracovat s rastrovým obrazem. Canvas umožňuje přes JavaScript načíst obraz a rozdělit jej na jednotlivé pixely, přesněji na jednotlivé RGBA složky. Tyto složky poté umožňuje analyzovat a měnit.

V tuto chvíli bylo nutno si uvědomit, jakým způsobem bude algoritmus mazání fungovat. Snažil jsem se vyhledat vysokoškolské práce s touto tématikou a našel jsem mnoho algoritmů, které jsou velmi široce využívány v grafických programech, jako je například tzv. kouzelná hůlka označující pixely na sebe napojené s určitou přednastavenou tolerancí rozdílu hodnot pixelů. Jakožto grafik však dobře vím, že tento nástroj není ideální pro ořezávání a mělo by se mu vyhnout.

V praxi se mi pro ořezávání produktů na světlém pozadí nejvíce osvědčilo využití masky za použití odstínů šedi. Tato maska určuje, které pixely budou viditelné (bílé), průhledné (černé), ale také mezihodnoty pomocí odstínů šedi. Bohužel JavaScript spolu s canvas elementem neumožňují kompoziční mód na bázi černobílé masky.

Toto mne vedlo k rozhodnutí, využití logiky masky, avšak rovnou ve výpočtu každého pixelu. Jednotlivé RGB hodnoty se zprůměrují a díky tomu získám odstín šedi. Tímto eliminuji možnost nepřesnosti, která by mohla nastat, pakliže bych kontroloval každou hodnotu RGB zvlášť. Toto by mohlo vést k rozdílné alfa hodnotě u dvou stejně světlých pixelů, které by však byly jinak barevné.

```
//ctx je 2D context canvasu
ctx.drawImage(img, 0, 0, iWidth * zoomX, iHeight * zoomY);
var imageData = ctx.getImageData(0, 0, iWidth, iHeight);
ctx.putImageData(imageData, 0, 0);
var data = imageData.data;
for (var y = 0; y < iHeight; ++y) {
    for (var x = 0; x < iWidth; ++x) {
        var index = (y * iWidth + x) * 4;
        var avg = (data[index] + data[index + 1] + data[index + 2]) / 3;
        //vUi udrzuje hodnoty range slideru
        if (avg <= vUi[0]) {
            data[index+3] = 255;
        } else if (avg > vUi[0] && avg <= vUi[1]) {
            data[index+3] = avg;
        } else {
            data[index+3] = 0;
        }
    }
}
```

```
    }  
}  
ctx.putImageData(imageData, 0, 0);
```

Výpis 2: Práce s RGBA hodnotami v canvas za pomoci JavaScriptu

Tento druh algoritmu však potřebuje uživatelskou korekci ve formě vyplnění obrysu objektu za pomoci vkládání geometrických tvarů, které budou vyplňovat obrysy vytvořené touto metodou. Objekty, ať už se bude jednat o kruhy, obdélníky či uživatelem vytvořené polygony bude nutno ukládat v paměti a to buď ve formě bitmapy, nebo polem samostatných JavaScriptových objektů. Tento postup je nutný vzhledem k častému překreslování obrazu, který bude překrývat dříve vykreslené pixely. Překreslení bude nutno provádět při každé změně v uživatelském rozhraní, což může být náročné na výpočetní výkon. V plánu je zpracovávat malé fotografie o velikosti maximálně 500 pixelů šířky na 500 pixelů výšky, tudíž, by výpočet a následné vykreslení v realtime neměl být problém. Pakliže by se jednalo o větší data bylo by nutné překreslovat pouze po tom, co by uživatel hodnoty potvrdil.

Pro ovládání obrazu bylo zapotřebí použít, nějaký druh UI, kterým by uživatel definoval, jaké hodnoty chce z obrazu vymazat za pomoci alfa kanálu. K tomuto bylo možno použít HTML input elementu, který však nedovoluje udržovat více než 1 hodnotu. Jediným možným způsobem jak zachovat UI v nativním HTML input rozhraní, bylo použití dvou input range sliderů. Tato možnost by však byla nejspíše matoucí, jak pro uživatele, tak pro použití v kódu. Vyžadovalo by to více event listenerů a zbytečný zmatek v kódu. Alternativou bylo využití jQuery UI a jeho slideru, který umožňuje získat více než jednu hodnotu. To znamená, že pokud by se algoritmus pro ořez změnil a přibyla by potřeba dalších hodnot, nebyl by problém takového rozhraní pozměnit. Nevýhodou této technologie byla však nutnost využití jak kódu pro jQuery UI, ale také samotného frameworku jQuery, který je potřeba pro funkci UI komponent. To znamená stažení většího množství kódu. Tuto potřebu jsem se snažil omezit alespoň využitím CDN zdroje, který by měl umožnit uživatelům, kteří již navštívili web využívající jQuery kód z této CDN, aby nemuseli znova stahovat data, ale využili ty, které má jejich prohlížeč v paměti.

4.8.3 Rozvoj

Do budoucna bych rád naimplementoval zvětšování za pomoci event listeneru naslouchajícímu scroll funkci, vracejícího x,y souřadnice. Pomocí funkce scale a translací bych obraz zvětšoval podle lokace kurzoru. Mám vše potřebné pro implementaci této funkcionality nastudováno, bohužel se však z nedostatku času musí tato implementace počkat.

Pro vylepšení algoritmu bych rád přidal možnost posunu viditelného okraje o uživatelem zadaný počet pixelů dále okolo neviditelného pixelu. Což by umožňovalo zbavit se nevzhledných artefaktů okolo hran produktů při práci s obrazem, který obsahuje silný anti-aliasing, či velkou kompresi.

4.8.4 Problémy

Jak jsem již výše psal, problém může nastat u většího obrázku, či pomalejšího počítače, vzhledem k tomu, že JavaScript je jazyk běžící na klientské straně. Dalším problémem jsou velmi světlé produkty splývající s pozadím. V takovýchto případech bude potřeba ručního ořezu za pomoci polygonového tělesa, či ořezu v externím grafickém programu.

4.9 IT pro Byznys

4.9.1 Zadání

Tento web má nabízet služby firmy AutoCont, avšak ve srozumitelné formě pro laiky. Grafický návrh jak tomu bylo u většiny projektů, dodalo grafické studio Aetna, podle návrhu našeho Ředitele pro portfolio a marketing. Návrh byl klasický jednosloupcový layout v určitých sekcích dělený do více boxů. Požadavkem byla možnost psaní článku, zobrazování nejnovějších článků, ať už se jedná o částečný výpis či plný, vyhledávání, odesílání emailů a udržování emailových adres, přidávání osobních zkušeností zákazníků a přihlašování. Další funkcionality by byla dodávána časem. Projekt byl svěřen mému novému kolegovi, který se zatím zaučuje. Mým úkolem bylo mu pomoci s úkoly, na které nestačí a poradit jak komplexně řešit vývoj tohoto webu a to jak ve front-endu tak na serverové straně.

4.9.2 Technologická rozhodnutí

Ještě před dodáním grafického návrhu jsme s mým novým kolegou měli na starost připravení back-end funkcionality, na kterou měl být později nasazen dodaný design. Pro získávání obsahu z webu bylo možné použít základní ActionResult funkcionalitu, která by vracela View, nebo využít asynchronního Ajax požadavku pomocí JavaScriptu a metod obsluhujících požadavky na back-endu. Já osobně jsem byl velkým zastáncem řešení pomocí Ajax, vzhledem k velikosti stránky a uživatelského pohodlí. Můj kolega však preferoval standardní postup za pomoci generování Views. Vzhledem k nativnímu předávání parametrů v ASP.NET MVC za pomoci url adresy bylo potřebovat přeměňovat adresy tak, aby odpovídaly uložení dat ve složkách na serveru a jménu dané stránky.

4.9.3 Problémy

Problém nastal při dodání grafického návrhu. V tento okamžik se změnila jak struktura designu a potřeba technologií pro zprovoznění tohoto designu, tak i samotné zadání požadavku na vnitřní řešení webové stránky. Z jednosloupcového layoutu se stal několika sloupcový layout s měnícím se počtem sloupců.

Dále zde bylo také excesivní množství reorganizace obsahu a to zvláště v hlavičce, kde se jednalo asi o 5 různých rozložení designových komponent, které byly naprosto přeorganizovány a nebylo tak možné docílit kýženého efektu bez použití absolutního pozicování a JavaScriptu.

Pro reorganizaci sloupců obsahu bylo využito JavaScriptového pluginu BlockIt.js, který řešil změny v pořadí animací. Toto je z psychologického hlediska velmi důležité, vzhledem k tomu, že krokové přeskakování obsahu přes sebe může uživatele velmi snadno znechutit, díky vyvolání pocitu zmatku a ztráty orientace.

Řešení vytváření a udržování obsahu za pomoci námi navržené jednoduché administrace bylo vedením zamítnuto. Bylo rozhodnuto o tom, že se nyní vytvoří pouze prototyp webu a později se předělá tak, aby běžel na redakčním systému Kentiko. Proti tomuto řešení jsem měl silné výhrady, vzhledem k tomu, že tato stránka nepotřebuje ve svém rozsahu takto silný nástroj na správu obsahu. Bohužel, má námitka byla zamítnuta a web poběží stejně jako stránka fóra autocontu na systému Kentiko. Bude tak potřeba nastudovat práci v tomto systému a naučit se vytvářet template designy pro tuto platformu. Byly nám přislíbeny školení na toto téma, což mne velmi potěšilo.

5 Časová náročnost projektů

Rád bych zde poznamenal, že časová náročnost je čistě orientační, protože na všech projektech se stále pracuje. Pořadí projektů je určeno podle toho, kdy nám bylo dodáno zadání, jelikož se na projektech pracovalo paralelně.

Tabulka 1: Tabulka časové náročnosti jednotlivých projektů

Název projektu	Počet dní
Web AutoCont CZ/SK/EN	9
eKatalog	5
Charitativní projekt AC25	3
SharePoint	2
Registrace na akce firmy AutoCont CZ i SK	4
AC Cloud	8
eKatalog Management Studio	12
Image Editor	4
IT pro Byznys	3

6 Znalosti

6.1 Znalosti načerpané při studiu na VŠB-TUO uplatněné v praxi

Při práci na projektech jsem obzvláště využil znalosti načerpané v předmětech „Vývoj informačních systémů“ a „Vývoj internetových aplikací“. Konkrétně se jednalo povědomí o systému TFS, práce s MVC, Visual Studio a frameworkem .NET, základní znalosti jQuery spolu s HTML a CSS.

6.2 Chybějící znalosti při nástupu do firmy

Vědomosti, které mi znatelně chyběly při působení ve firmě byly zejména neznalost CSS pseudo-prvků jako je :before a :after a jejich praktické využití. Neznalost CSS procesorů, Bootstrapu, hlubší znalosti JavaScriptu, jako je vytváření scopů a strukturování, jQuery a úplná neznalost jQuery UI. Dále by se mi velmi hodily znalosti vnitřní konfigurace webu (webconfig) a nuget manageru.

7 Závěr

Jsem velmi vděčný za možnost absolvování bakalářské práce formou praxe, kterou mi VŠB-TUO nabídla. Do firmy AutoCont CZ jsem nastoupil, dalo by se říci prakticky neznalý nových technologií v oboru vývoje webových stránek, ale odešel jsem jako člověk připravený do pracovního procesu znalý mnoha užitečných technologií, jazyků a postupů. Dále jsem velice vděčný firmě AutoCont CZ a mému týmu, že mne přijali a věnovali mi svůj čas a znalosti. Naučil jsem se mnoho nového a znalosti, které jsem měl, jsem dopiloval. Firmou jsem byl podporován v mém rozvoji a bylo mi umožněno studovat zajímavé technologie, které jsem později mohl i využít v praxi, ale je toho ještě mnoho, co se chci naučit a to i včetně WebGL. Bylo mi umožněno vybrat si školení z nabídky našeho školicího střediska, které plánuji v brzké době uplatnit. Tato praxe mne v mém osobním i pracovním životě posunula velmi daleko a jsem opravdu rád za to, že jsem si vybral tuto formu bakalářské práce.

Literatura

- [1] W3Schools [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://www.w3schools.com/>
- [2] Stackoverflow [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://stackoverflow.com/>
- [3] YouTube - kanál kudvenkat [online]. 2016 [cit. 2016-04-28]. Dostupné z: <https://www.youtube.com/user/kudvenkat/playlists>
- [4] Codrops [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://tympanus.net/codrops/2015/02/26/inspiration-button-styles-effects/>
- [5] Bootstrap [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://getbootstrap.com/components/>
- [6] CSS-Tricks [online]. 2016 [cit. 2016-04-28]. Dostupné z: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- [7] AutoCont [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://www.autocont.cz/o-spolecnosti>
- [8] Vzhůru dolů [online]. 2016 [cit. 2016-04-28]. Dostupné z: <http://www.vzhurudolu.cz/blog/15-css-preprocesory-4>